REMARKS

Claims 1-2, 4-11, 13-19 and 21-23 are pending in the present application. Claims 1, 10, and 19 were amended. Claim 24 was added. Claims 3, 12, and 20 were canceled. Support for the amended claims and the newly added claim can be found in the Specification at least on page 15, lines 17-24, page 17, lines 24-26, and page 18, lines 4-17. Reconsideration of the claims is respectfully requested.

Amendments were made to the specification to correct errors and to clarify the specification. No new matter has been added by any of the amendments to the specification.

I. Examiner Interview

Applicant thanks Examiner Maniwang for all the courtesies extended Applicant's representative during the March 28, 2006 telephone interview. During the interview, Applicant's representative discussed the prior art of record and the manner in which *Apfel* and *Douvikas* fail to teach or disclose the features recited in the presently claimed invention in independent claims 1, 10, and 19. The Examiner indicated that he would consider the arguments and amendments when submitted. The arguments discussed as well as additional reasons that the claims are not anticipated are set forth in the remarks below.

II. 35 U.S.C. § 102, Anticipation: Claims 1-2, 5-11, 14-19 and 22-23

The examiner has rejected claims 1-2, 5-11, 14-19 and 22-23 under 35 U.S.C. § 102(e) as being anticipated by *Apfel* et al., System and Method for Creating and Inserting Multiple Data Fragments into an Electronic Mail Message, U.S. Patent No. 6,510,453, January 21, 2003 (hereinafter "*Apfel*"). This rejection is respectfully traversed.

The examiner states in the Office Action that:

> 6. Regarding claims 1, 10, and 19, *Apfel* disclosed a method and system for providing personal data to a recipient comprising providing a personal data object, wherein the personal data object includes personal data and a template with embedded code for generating a personal data output (see column 2, lines 1-30, column 14, lines 28-36) receiving a credential for the recipient (see column 14, lines 44-52) activating the embedded code in the template to dynamically generate a personal data output based on the at least one credential (see column 2, lines 17-21 column 3, lines 43-55, column 14, lines 44 52) and delivering the personal data output to the recipient (see column 15, lines 13-26).

Office Action dated December 29, 2005, page 3.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the

claims. In re Bond, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. In re Lowry, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. Kalman v. Kimberly-Clark Corp., 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). In this particular case, the cited reference does not teach or disclose each and every feature of the presently claimed invention as they are arrange in the claims.

1. **Independent Claims 1, 10, and 19**

Claim 1, as amended, recites:

> 1. A method for providing personal data to a recipient, comprising:
> providing a personal data object, wherein the personal data object includes personal data and a template with embedded program code for generating a personal data output;
> receiving, from the recipient, at least one credential for the recipient;
> activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential; and
> delivering the personal data output to the recipient.

Independent claims 10 and 19 recite similar subject matter as is recited in independent claim 1. The steps for "providing a personal data object" that includes personal data and a template with embedded program code for generating a personal data output; "receiving, from the recipient, at least one credential from the recipient;" and "activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential," recited in claim 1 are missing from the cited prior art.

a. **Providing a personal data object**

Apfel does not disclose "providing a personal data object, wherein the personal data object includes personal data and a template with embedded program code for generating a personal data output," as is recited in claim 1. The Examiner has cited *Apfel* at column 2, lines 1-30 and column 14, lines 28-36 as disclosing this feature. The cited portion of *Apfel* at column 2, lines 2-30 states:

> The present invention supports the automated insertion of data into an electronic document based on an identifying characteristic of the electronic document. For the representative example of a electronic mail message, the present invention supports the insertion of an electronic mail signature fragment into an electronic mail message based on the context of the electronic mail message. If the electronic mail message represents a reply to a received message, then a signature fragment assigned to reply-type messages is inserted without manual intervention in a predetermined location within the electronic mail message. Alternatively, if the electronic mail message represents a newly composed message, then a signature fragment assigned to new-type messages is

> inserted without manual intervention into the predetermined location within the electronic mail message. The present invention also can support the automated insertion of a particular signature fragment within the electronic mail message based on the identity of the designated recipient for the message.
>
> In general, the present invention can insert one of multiple signature fragments into a electronic mail message, typically at the close of the message, in response to identifying the context or type of the message. Based on this determination of the message type, a data set, such as a data string representing one of the signature fragments, is selected for insertion into the message. In turn, this data string can be inserted at a predetermined location within the message to complete an initial composition of the message.

Apfel, column 2, lines 2-30.

This portion of *Apfel* discloses the automated insertion of data, such as a signature fragment, into an email. A particular signature fragment is inserted based on the type of email message being sent. For example, if a new email is being sent, a signature assigned to new messages is inserted. If an email is being sent in reply to a received email, then a signature fragment assigned to reply messages is inserted. A signature fragment can also be inserted based on the designated recipient for the message. Thus, *Apfel* merely teaches inserting a signature fragment into an email based on the type of email in order to complete composition of the email message.

However, *Apfel* does not teach or even mention a personal data object that includes personal data and **a template with embedded program code for generating a personal data output**. Although *Apfel* might teach inserting an assigned and, presumably pre-generated or pre-created, signature fragment into an email message, such teachings are not the same as **a template** with embedded program code **for generating a** personal data output. Inserting a pre-created signature fragment retrieved from a storage in no way discloses generating personal data output by a template with embedded code that is included in a personal data object. In fact, the cited portion of *Apfel* does not mention a data object, a template, or embedded code of any type anywhere in this portion of the reference.

Apfel also states:

> Turning now to FIG. 7, the association of custom electronic mail signature fragments 214 with specific electronic mail recipients will be discussed. That is, each recipient of electronic mail sent through the use of an exemplary embodiment may have a custom signature fragment 214 associated with him or her. This allows a user of an exemplary embodiment to specify more than simply a default new message signature fragment and a default reply message signature fragment, thus leading to increased flexibility.
>
> Generally, a list of all recipients of electronic mail messages 212 is maintained. Further, this list contains the electronic signature fragment 214 used in the electronic mail message 214 sent to that recipient. A typical list is thus

> comprised of rows, which in turn have two entries per row: a recipient name (or other unique identifier) and the signature fragment associated with that recipient.
>
> In step 700, when an electronic mail message 214 is composed and a recipient selected in the "TO" field 204, the name of the recipient is checked against the stored list of recipients. If a match is found, then step 702 is accessed. In step 702, the name of the electronic mail signature fragment 214 associated with the recipient of the electronic mail message 212 is retrieved from a row of the list of recipients. Once the associated signature fragment is known, step 704 is entered.
>
> In step 704, the data of the associated electronic mail signature fragment 214 is retrieved from storage.

Apfel, Column 14, lines 28-54.

Here, *Apfel* describes associating a custom email signature fragment, other than the default new email signature fragment and default reply signature fragment, with a specific recipient. *Apfel* discloses maintaining a list comprising rows. Each row has a recipient name entry and a signature fragment associated with the recipient. After an email message is composed and an email recipient selected, the recipient is checked against the list. If a match is found, the signature fragment associated with the recipient is retrieved from storage. Once again, *Apfel* merely discloses inserting a user created signature fragment into an email message rather than providing for **generating** a personal data output by a program code embedded in a template. In fact, *Apfel* states that the signature fragment is **retrieved from storage** rather than dynamically generated by program code, as is claimed in claim 1. Thus, the cited reference clearly does not teach this feature. Likewise, the cited portion of the reference does not even mention a data object or a template having embedded code of any kind. Moreover, the reference fails to teach or even mention a personal data object that includes personal data and a **template with embedded program code** for generating a personal data output in this or any other section of the reference.

b. Receiving at least one credential

Apfel does not teach "receiving, from the recipient, at least one credential for the recipient," as is claimed in amended claim 1. The Examiner cites to *Apfel* at column 14, lines 44-52, which is quoted above. As discussed above, this portion of the reference merely teaches retrieving a signature fragment from storage based on an association with a recipient name in a list. The cited portion of the reference does not teach **receiving** at least one credential.

The claim has been amended to make it clear that the at least one credential is received from the recipient. The cited portion of the reference does not teach or even mention receiving anything from the recipient. Although this portion of the reference may disclose retrieving a signature fragment from storage, the signature fragment is not a credential for the recipient that is received from the recipient.

Moreover, even if the recipient name entered into the "TO" field of an email by a user composing an email is inherently a credential, such teachings merely disclose entering a recipient name rather than **receiving at least one credential** for the recipient. Therefore, the cited portion of the reference fails to teach "receiving, from the recipient, at least one credential for the recipient," as is claimed in amended claim 1.

c. **Activating the embedded code in the template**

Apfel fails to disclose "activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential," as is claimed in independent claim 1. The Examiner believes this feature is disclosed by *Apfel* at column 2, lines 17-21; column 3, lines 43-55; and column 14, lines 44-52. The cited portion of *Apfel* at column 2, lines 17-21 is quoted above. As shown above, this section of *Apfel* merely discloses the automated insertion of a signature fragment into an email based on the identity of a recipient of the email message. As discussed above, inserting a signature fragment that was retrieved from a storage does not disclose activating an embedded program code in a template to **dynamically generate** a personal data output. Moreover, inserting a signature fragment based on an identity of the designated recipient of the email message does not teach dynamically generating a personal data output **based on the at least one credential** that was received from the recipient. Merely stating that a signature fragment is inserted based on an identity of a recipient does not explicitly or inherently disclose at least one credential for a recipient. Moreover, even if an identity of the designated recipient does inherently disclose a credential for the recipient, the identity of the recipient is designated in the "TO" field of the email rather than received from the recipient.

The cited portion of *Apfel* at column 3, lines 43-55 states:

> The present invention also can support the linking of signature files and recipients. That is, when a particular recipient is specified for an electronic mail message, a signature designated for that recipient will be inserted into the body of the message rather than the default new or reply signature fragments. This allows for greater personalization of e-mail messages without the expenditure of additional effort by the user. To support this feature of the invention, a table of all previous recipients, called the association list, can be maintained within memory; this table typically comprises a set of entries, a unique recipient identifier and the electronic mail signature fragment associated with that recipient.

Apfel, Column 3, lines 42-55.

This section of *Apfel* discloses associating a particular signature fragment/file with a particular recipient. The associated signature fragment is inserted into emails to that recipient rather than the default new signature or default reply signature fragment. An association list contains a unique recipient

identifier and the signature fragment associated with that recipient. Once again, although *Apfel* may disclose inserting a signature fragment into an email based on a recipient, such teachings do not disclose **dynamically generating** a personal data output **based on the at least one credential** that was received. In fact, *Apfel* teaches that the signature fragment is created by a user and stored for later retrieval, rather than dynamically generated by embedded code in a template. *Apfel* states:

> Alternately, the decision may be made in step 352 to create a new electronic mail signature fragment 214. If so, then in step 354 the text of the signature fragment 214 is inputted in editing window 306. This text may be formatted as desired through the use of editing toolbar 308. Editing options include, but are not limited to, changing the font, size, justification, color, and style of the signature fragment 214. The editing toolbar 308 is also available for use with a previously stored electronic mail signature fragment 214 that has been selected for editing, as described above.
>
> Once the electronic mail signature fragment 214 has been created or edited in step 354, a name is assigned in step 356 to the signature fragment. The name of the signature fragment is inputted into the name window 302. This is the name by which the signature fragment 214 is stored, and later displayed as part of a list of available signature fragments as described with respect to step 350.
>
> Next, in step 358 the signature fragment 214 is stored for later use and retrieval.

Apfel, paragraph 9, lines 45-63.

As shown above, *Apfel* discloses that a signature fragment is created by a user inputting text in an editing window. The signature fragment is assigned a name and stored for later use and retrieval. In contradistinction, the presently claimed invention in claim 1 **dynamically generates** a personal data output by activating **embedded program code in the template.**

Finally, the Examiner also cites to column 14, lines 44-52, which is quoted above. As discussed above, this portion of *Apfel* merely describes checking a recipient listed in a "TO" field of an email with the list of recipient names and associated signature fragments. If a match is found, the associated signature fragment is retrieved from the list and inserted into the email. As shown above, *Apfel* does not teach embedded code in a template or dynamically generating a personal data output based on at least one credential. Moreover, *Apfel* does not teach or even mention embedded code in a template for dynamically generating a personal data output based on at least one received credential in any other section of the reference. Thus, *Apfel* fails to teach activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential," as is recited in claim 1.

Apfel fails to teach each and every feature recited in independent claim 1. Moreover, other rejected independent claims 10 and 19 recite subject matter addressed above with regard to claim 1.

Therefore, claims 10 and 19 are distinguishable over *Apfel* for the same reasons set forth above with regard to claim 1.

2. Dependent Claims 2, 5-9, 11, 14-18, 22, and 23

Since dependent claims 2, 5-9, 11, 14-18, 22, and 23 depend from independent claims 1, 10 and 19, the same distinctions between *Apfel* and the claimed invention in dependent claims 2, 5-9, 11, 14-18, 22, and 23 are applicable to these claims. Therefore, at least by virtue of their dependency on independent claims 1, 10 and 19 dependent claims 2, 5-9, 11, 14-18, 22, and 23 are allowable over the prior art.

Moreover, dependent claims 2, 5-9, 11, 14-18, 22, and 23 recite additional combinations of features that are not taught or suggested by the cited prior art. For example, regarding claims 2, 11, and 22, the Examiner believes *Apfel* discloses the credential comprising an email address at column 2, lines 18-21 and column 14, lines 44-52, which are quoted above. As discussed above, the recipient discloses in *Apfel* is designated in the "TO" field of an email, rather than received. Thus, the email recipient designated in *Apfel* is not the same as the claimed credential comprising an email address. Therefore, *Apfel* fails to teach each and every feature of claims 2, 11, and 22.

Regarding newly added claim 24, *Apfel* fails to teach the credential comprises a device ID. *Apfel* merely teaches inserting a signature fragment into an email based on a designated recipient. *Apfel* does not teach or mention an identity of a device. Therefore, *Apfel* fails to teach each and every feature of claim 24. Therefore, the rejection of claims 1-2, 5-11, 14-19 and 22-23 under 35 U.S.C. § 102(e) has been overcome.

Furthermore, *Apfel* does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. *Apfel* actually teaches away from the presently claimed invention because it teaches inserting a signature fragment retrieved from a storage as opposed to a dynamically generating a data output by an embedded code in a template as in the presently claimed invention. Absent the examiner pointing out some teaching or incentive to implement *Apfel* and dynamically generating personal data output by an embedded code in a template, one of ordinary skill in the art would not be led to modify *Apfel* to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify *Apfel* in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the applicants' disclosure as a template to make the necessary changes to reach the claimed invention.

III. 35 U.S.C. § 102, Anticipation: Claims 1-2, 4-11, 13-19 and 21-23

The examiner has rejected claims 1-2, 4-11, 13-19 and 21-23 under 35 U.S.C. § 102(e) as being anticipated by *Douvikas* et al., E-Service to manage and Export Contact Information, U.S. Patent No. 6,633,311, October 14, 2003 (hereinafter "*Douvikas*"). This rejection is respectfully traversed.

1. Independent Claims 1, 10, and 19

The examiner states in the Office Action that:

> 14. Regarding claims 1, 10, and 19 Douvikas disclosed a method and system for providing personal data to a recipient comprising proding a personal data object (see column 8, lines 7-17), wherein the personal data object includes personal data (see column 8, lines 52-53) and a template with embedded code for generating a personal data output (see column 13, lines 15-22) receiving a credential for the recipient (see column 9 lines 21-56), activating the embedded code in the template to dynamically generate a personal data output based on the at least one credential (see column 101 lines 58 64 column 13, lines 15-17), and delivering the personal data output to the recipient (see column 8, lines 14-17, column 10, lines 48 51).

Office Action dated December 29, 2005, page 4.

Douvikas fails to teach "providing a personal data object" that includes personal data and a template with embedded program code for generating a personal data output; "receiving, from the recipient, at least one credential for the recipient;" and "activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential," as is recited in amended claim 1.

a. Providing a personal data object

Douvikas does not disclose "providing a personal data object, wherein the personal data object includes personal data and a template with embedded program code for generating a personal data output," as is recited in claim 1. The Examiner has cited *Douvikas* at column 8, lines 7-17, column 8, lines 52-53, and column 13, lines 15-22 as disclosing this feature. The cited portion of *Douvikas* at column 8, lines 7-17 is included within the following section of *Douvikas* which states:

> Either the signature hyperlink or the vCard (which can also contain a hyperlink) can then be used by conventional email programs. Electronic mail sent by the cardholder is automatically formatted to contain a signature hypertext link, according to the well-known hypertext markup language (HTML) standard or any of its common variants, directing recipients of the email to the electronic business card access and organization system. This hyperlink enables the recipient of the email to rapidly access the EBC system to locate the cardholder and/or obtain additional information. In effect, receipt of an email containing the hyperlink enables the recipient to easily become a user. In some embodiments, the signature hyperlink is part of the vCard feature known and implemented in

> common email programs such as Microsoft Outlook and Netscape Communicator.RTM. In an alternate embodiment, the signature hyperlink is implemented using the well-known email signature block feature.

Douvikas, Column 8, lines 7-24.

This portion of *Douvikas* discloses an email sent to a recipient that contains a link to an electronic service that provides access to contact information. Although *Douvikas* discloses providing a link to connect a recipient to the service, this portion of *Douvikas* does not disclose providing **a personal data object that includes a template with embedded program code** for generating a personal data output. In fact, this section of *Douvikas* does not even mention a personal data object or a template with embedded program code of any kind

The cited portion of *Douvikas* at column 8, lines 52-53 is included in the section of *Douvikas* that states:

> Let's walk through the process of becoming a Member.
> 1. From the Member login screen, click the Become a Member button.
> 2. Fill in your Card Profile: the profile contains all of your contact information and can be updated as needed. See the help menu topic "Set Up Your Card" for more information.
> After your membership is confirmed, you can log in to ecardfile.com using your Card ID and password. After log in, you are brought to your personal ecardfile area. Here is where you can store other Member cards and perform functions such as adding, deleting, changing the privacy level access to your Card that you have given to other Members, and exporting a card to your address book.

Douvikas, Column 8, lines 44-62.

Here, *Douvikas* discloses the process of becoming a member of the electronic service that provides access to contact information. A user fills out a profile containing the users contact information. After becoming a member, the user can add, delete or change privacy levels for the user's own contact information and store contact information (member cards) for other members of the service. *Douvikas* merely teaches becoming a member of a service in order to gain access to contact information available from the service, as well as to make the user's own contact information available to other users of the service. At best, *Douvikas* teaches that a vCard can contain a hyperlink. However, such teachings do not disclose a **personal data object that includes a template with embedded program code** for **generating** a personal data output. As can be seen above, the cited portion of the reference does not even mention a data object, a template, or generating personal data output. In fact, *Douvikas* teaches accessing information stored by the service rather than **generating** personal data as is claimed in claim 1.

The Examiner also cites to *Douvikas* at column 13, lines 15-22, which states:

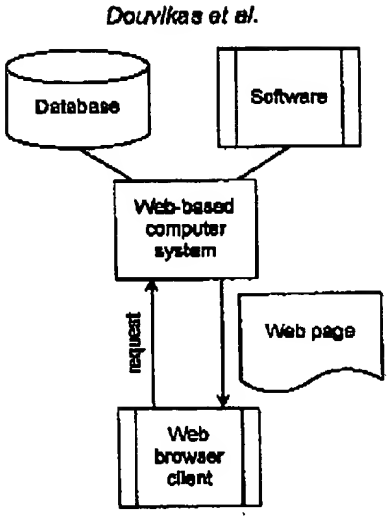
> All pages displayed by the Boomerang application, including the help and information screens, are dynamically generated. The base HTML code and image links for these pages are stored as template files which are preloaded on servlet initialization. These files are parsed and custom tags replaced with data extracted from the database (or calculated) before sending the page to the requester and display to the user.

Douvikas, Column 13, lines 14-22.

This portion of the reference refers to utilization of a template to display help and information screens. Template files are preloaded at servlet initialization, parsed, and custom tags are replaced with data from a database. Thus, *Douvikas* merely teaches a template with embedded tags for generating display help and information screens. *Douvikas* does not explicitly or inherently teach utilization of a template to generate personal data output based on at least one credential. Moreover, even if this section of the reference does inherently disclose utilization of a template to display contact information, *Douvikas* still does not teach a template with **embedded program code** for generating personal data output **based on the at least one credential.** In fact, *Douvikas* teaches using specific software, separate from the data itself to manage access and data privacy, as opposed to a personal data object that includes a template with **embedded** program code, as in the presently claimed invention. *Douvikas* teaches a front end software framework that manages access to electronic business card data. The front end is separate and distinct from the business data itself. *Douvikas* states:

> In one embodiment of the present invention, an electronic business card (EBC) access and organization system operates from a Web-based computer system that includes a database and software for managing access, data privacy, and dynamic updates. The cardholder database, i.e., the database containing records of each registered cardholder (or "Member" of the EBC system), is accessible from any Web browser connected to the Internet. Examples of such common Web browsers are Microsoft's Internet Explorer and Netscape.RTM. Navigator.RTM.. In an alternate embodiment, the EBC system may be installed behind a conventional network "firewall" security device and thus made accessible only to browsers connected to and authorized to use the intranet defined by and behind the firewall.

Douvikas, col. 2, lines 17-30. Below is a block diagram of the EBC system as described by *Douvikas*:



As shown above, a Web-based computer system includes a database and software. The database and the software are separate, as described in *Douvikas*. Also, the software, not the database, is responsible for managing access and data privacy. Contrary to the present invention, in no way is the software of *Douvikas* **embedded** in a personal data object.

Thus, *Douvikas* fails to teach "providing a personal data object, wherein the personal data object includes personal data and a template with embedded program code for generating a personal data output," as is claimed in claim 1.

b. **Receiving at least one credential**

Douvikas does not teach "receiving, from a recipient, at least one credential for the recipient," as is claimed in claim 1. The Examiner believes this feature is discloses by *Douvikas* at column 9, lines 21-56, which states:

> As you are entering information into your Card Profile, please keep in mind that ecardfile.com gives you three levels of privacy for each field:
>
> Level 1-Public. Information at this level will be displayed to anyone who looks up your card. This could be anyone viewing cards from the World Wide Web, whether you know them or not.
>
> Level 2-Semi-Private. Information at this level will displayed only to other ecardfile Members who are in your personal ecardfile and who have been designated to receive your semi-private information.
>
> Level 3-Private. Information at this level will be displayed only to other ecardfile Members who are in your personal ecardfile and who have been designated to receive your private information.
>
> All field information is set to private when you first fill out a Card Profile. Be sure to select other privacy levels for fields that are either semi-private or public.

> The Email Auth field is used only by ecardfile.com for verification purposes. It is never displayed on your Card. You must enter a current email address in the Email Auth field. Once you complete the Card Profile and click "OK," ecardfile sends an email to this address and waits for your reply before authorizing your membership and enabling you to log in. This authorization process has been designed to protect your privacy and identity.
>
> Add Others' Cards to Your Ecardfile
>
> From your personal ecardfile screen, use the Look Up fields to view the card of the Member you want to add. When the card is displayed, press the Add icon.
>
> If you would like to give this Member access to your semi-private or private ecardfile information, be sure to change the privacy level displayed next to the Member's name. See the help topic "Set/change privacy levels" for more information.

Douvikas, Column 9, lines 22-56.

Here, *Douvikas* discloses a user setting a privacy level for each file in a user's card profile. Three privacy levels are disclosed. At level 1, all information is displayed to everyone. At level 2 and level 3, the information will only be displayed to another member designated by the user to receive the information. The privacy levels for controlling display of information are **designated by the user** to select which data to display rather than a **credential that is received from the recipient.** Moreover, *Douvikas* discloses that the user changes the privacy settings for the recipient, rather than receiving at least one credential from the recipient. Thus, *Douvikas* fails to disclose "receiving, from a recipient, at least one credential for the recipient," as is recited in claim 1.

c. **Activating the embedded code in the template**

Douvikas fails to disclose "activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential," as is claimed in independent claim 1. The Examiner believes this feature is discloses by *Douvikas* at column 10, lines 58-64 and column 13, lines 15-17. The cited portion of *Douvikas* at column 10, lines 58-64 is included in the section of *Douvikas* that states as follows:

> A signature file has an HTML link to your Card; when downloaded, the signature file will embed the link into all of your email messages. When someone reads your message and wants to view your contact information, he just clicks on the HTML link and is immediately connected to your Card and your up-to-the-minute contact information.
>
> A vCard is a file that holds your contact information in a standard format. Some email packages such as Microsoft Outlook and Netscape Communicator recognize this format and can treat it in a special way. Because it

> is not a live link, it may display old or inaccurate information, particularly if someone is reading an old email message from you.
>
> If your email package, or more importantly the message recipient's email package, does not support HTML tags or vCards, you may cut and paste the HTML link displayed and attach it to your messages. The recipient just clicks or cut and pastes the HTML link into a browser and is immediately connected to your Card and your up-to-the-minute contact information.

Douvikas, Column 10, lines 46-64.

This portion of *Douvikas* states that when a signature file is downloaded it creates a link to a user's card that is embedded in the user's email messages. A recipient of the email message can click on the link in order to be connected to the service and view the user's contact information. *Douvikas* merely teaches embedding a link in an email. Such statements do not explicitly or inherently teach a template with embedded program code to dynamically generate a personal data output. Moreover, even if the embedded link in an email for connecting to the service could inherently teach an email with embedded code for displaying contact personal data, such teachings would still fail to disclose **a personal data object** including a template with **embedded program code to dynamically generate** a personal data output, as opposed to merely displaying contact information. In fact, *Douvikas* discloses **displaying** contact information **based on a user designated privacy level** rather than generating personal data output **based on at least one credential** received **from the recipient.** Thus, as shown above, this section of *Douvikas* merely discloses the automated insertion of a link to the electronic service's web site rather than a personal data object with embedded code for generating personal data output. Furthermore, simply disclosing that a link to a service web site does not inherently disclose at least one credential for the recipient or generating personal data output based on the at least one credential.

The other cited portion of *Douvikas* at column 13, lines 15-17 is quoted above. As discussed above, this portion of the reference merely discloses template files for displaying help and information screens. The template files are parsed and custom tags are replaced with data from a database. This portion of *Douvikas* does not disclose a template with **embedded program code** for generating a personal data output **based on at least one credential** received from a recipient. Thus, *Douvikas* fails to teach activating the embedded program code in the template to dynamically generate a personal data output based on the at least one credential," as is recited in claim 1.

Douvikas fails to teach each and every feature recited in amended independent claim 1. Moreover, other rejected independent claims 10 and 19 recite subject matter addressed above with regard to claim 1. Therefore, claims 10 and 19 are distinguishable over *Douvikas* for the same reasons set forth above with regard to claim 1.

2. **Dependent Claims 2, 5-9, 11, 14-18, 22, and 23**

Since dependent claims 2, 5-9, 11, 14-18, 22, and 23 depend from independent claims 1, 10 and 19, the same distinctions between *Douvikas* and the claimed invention in dependent claims 2, 5-9, 11, 14-18, 22, and 23 are applicable to these claims. Therefore, at least by virtue of their dependency on independent claims 1, 10 and 19 dependent claims 2, 5-9, 11, 14-18, 22, and 23 are allowable over the prior art. Moreover, dependent claims 2, 5-9, 11, 14-18, 22, and 23 recite additional combinations of features that are not taught or suggested by the cited prior art.

For example, regarding claims 4, 13, and 21, *Douvikas* does not teach or suggest that the personal data object comprises at least one of a signed Java class, a Java server page, and a text file with fields replaced by JavaScript code. Moreover, even if *Douvikas* teaches that the electronic business card service software uses Java, the applied references does not teach a **personal data object** that includes personal data and a template with embedded code, wherein **the personal data object** comprises a signed Java class, a Java server page, and a text file with fields replaced by JavaScript code, as recited in claims 4, 13, and 21.

Regarding newly added claim 24, *Douvikas* fails to teach the credential comprises a device ID. *Douvikas* merely teaches a user designating a privacy level for the user's contact information. *Douvikas* does not teach or even mention receiving a device ID from a recipient and activating embedded code in the template to dynamically generate a personal data output based on the device ID. Therefore, *Douvikas* fails to teach each and every feature of claim 24. Therefore, the rejection of claims 1-2, 5-11, 14-19 and 22-23 under 35 U.S.C. § 102(e) has been overcome.

Furthermore, *Douvikas* does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. *Douvikas* actually teaches away from the presently claimed invention because it teaches using specific server software to query database records and fields, as opposed to a personal data object that includes a template with embedded code, as in the presently claimed invention. Absent the Office Action pointing out some teaching or incentive to implement *Douvikas* with a personal data object with embedded code, one of ordinary skill in the art would not be led to modify *Douvikas* to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify *Douvikas* in this manner, the presently claimed invention can be reached only through an improper use of hindsight using Applicants' disclosure as a template to make the necessary changes to reach the claimed invention.

Therefore, the rejection of claims 1-2, 4-11, 13-19 and 21-23 under 35 U.S.C. § 102(e) has been overcome.

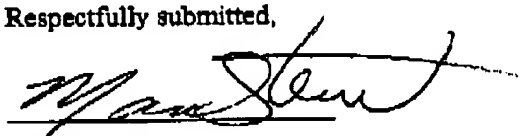
IV. **Conclusion**

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: March 28, 2006

Respectfully submitted,



Mari Stewart
Reg. No. 50,359
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicants